GENERALIZED COUNTING CONSTRAINT SATISFACTION PROBLEMS WITH DETERMINANTAL CIRCUITS

JASON MORTON[1] AND JACOB TURNER[1]

Abstract. Generalized #CSP problems include Holant problems with planarity restrictions; polynomial-time algorithms for such problems include matchgates and matchcircuits, which are based on Pfaffians. In particular, they use gates which are expressible in terms of a vector of sub-Pfaffians of a skew-symmetric matrix. We introduce a new type of circuit based instead on determinants, with seemingly different expressive power. In these *determinantal circuits*, a gate is represented by the vector of all minors of an arbitrary matrix. Determinantal circuits permit a different class of gates. Applications of these circuits include a new proof of the Chung-Langlands formula for the number of rooted spanning forests of a graph and a strategy for simulating quantum circuits with closed timelike curves. Monoidal category theory provides a useful language for discussing such counting problems, turning combinatorial restrictions into categorical properties. We introduce the counting problem in monoidal categories and count-preserving functors as a way to study FP subclasses of problems in settings which are generally #P-hard. Using this machinery we show that, surprisingly, determinantal circuits can be simulated by Pfaffian circuits at quadratic cost.

1. Introduction

Let $\mathrm{Vect}_{\mathbb{C}}$ be the category of finite-dimensional vector spaces and linear transformations over the base field $\mathbb{C}$. A string diagram [8] in $\mathrm{Vect}_{\mathbb{C}}$ is a *tensor (contraction) network*, and fixing such a diagram, the problem of computing the morphism represented is the *tensor contraction problem*, which is in general #P-hard (examples include weighted counting constraint satisfaction problems [4]).

We study complex-valued tensor contraction problems in subcategories of $\mathrm{Vect}_{\mathbb{C}}$ by considering them as diagrams in a monoidal category. For a survey of the rich diagrammatic languages that can be specified similarly see [18] and the references therein. By a *circuit* we mean a combinatorial counting problem expressed as a string diagram in a monoidal subcategory of $\mathrm{Vect}_{\mathbb{C}}$ (that is, a tensor contraction network). Such diagrams generalize weighted constraint satisfaction problems and Boolean circuits; for example fanout and swap are not necessarily available. As in the Boolean case, such circuits often have more familiar description languages. Subcategories of $\mathrm{Vect}_{\mathbb{C}}$ faithfully represent Boolean[11] and quantum circuits[1], counting constraint satisfaction problems, and many other problems[6].

Suppose we have a a problem $\mathcal{L}$, for example a counting constraint satisfaction problem [3] given by an embedded planar graph with variables or weighted constraints at each vertex. We can consider such a problem to be described by the

[1]Department of Mathematics, Pennsylvania State University, University Park PA 16802

Date: February 6, 2013.

data of a monoidal world (see e.g. [10], Chapter 12). We consider an *interpretation* [18] map $i : \mathcal{L} \to \mathrm{Vect}_{\mathbb{C}}$ that gives an interpretation of the problem of interest as a tensor contraction problem in $\mathrm{Vect}_{\mathbb{C}}$. A second, count-preserving functor h from a category $\mathcal{C}$ in which contraction is in FP and a subcategory $\mathcal{S}$ of $\mathrm{Vect}_{\mathbb{C}}$ serves to characterize the problems which can be solved in polynomial time according to a particular contraction scheme. Many such schemes (such as holographic algorithms [19]) work by exploiting some combinatorial identity or *kernel* relating an exponential sum (corresponding to performing the tensor contraction by a naïve algorithm) and a polynomial time operation (often the determinant or Pfaffian of a matrix) that yields the same result. They can be viewed as a complementary alternative method to geometric complexity theory [17] in the study of which counting problems (such as computing a permanent) may be embedded in a determinant computation at polynomial cost.

We formulate a class of circuits based on determinants and show that it is always solvable in polynomial time. This existence of such a class was conjectured in [12]. We give some applications of these circuits including a new proof of the Chung-Langlands formula [5] for the number of rooted spanning forests of a graph. We explain the relationship between Pfaffian circuits [15] (and so matchgates) and determinantal circuits and prove a functorial relationship between them. We show that, surprisingly, every determinantal circuit can be expressed as a Pfaffian circuit at quadratic cost.

This paper is organized as follows. In Section 2, we describe the counting problem in monoidal categories and the setting for our results. In Section 3 we define determinantal circuits and give applications to the Chung-Langlands rooted spanning forest theorem (Section 4) and quantum circuits with postselection-based closed timelike curves (P-CTC, Section 4.3). In Section 5 we relate determinantal and Pfaffian circuits.

2. Toward a categorical formulation of counting complexity

Let $\mathcal{M}$ be a (strict) monoidal category [14] with monoidal identity $\mathbb{1}_{\mathcal{M}}$ and such that $S_{\mathcal{M}} = \mathrm{Hom}_{\mathcal{M}}(\mathbb{1}_{\mathcal{M}}, \mathbb{1}_{\mathcal{M}})$ is a semiring; call this a *semiringed category.* A *monoidal word* is a collection of morphisms composed and tensored together to form a new morphism [10]. The problem of *counting* in a semiringed category $\mathcal{M}$ is to determine which morphism in $\mathrm{Hom}_{\mathcal{M}}(\mathbb{1}_{\mathcal{M}}, \mathbb{1}_{\mathcal{M}})$ is represented by an arbitrary monoidal word in $\mathcal{M}$ with $\mathbb{1}_{\mathcal{M}}$ as its domain and codomain.

Example 2.1. Consider the monoidal category in which the objects are cartesian powers of a two-element set $B = \{T, F\}$. For example

$$\{T, F\}^2 = \{(T, T), (T, F), (F, T), (F, F)\}$$

which we denote by $\{T \otimes T, T \otimes F, F \otimes T, F \otimes F\}$ since the tensor product of objects in this category is the cartesian product of the corresponding sets. Morphisms are nonnegative integer valued binary relations such as $\{2 \cdot (T \otimes F, T \otimes T), 3 \cdot (T \otimes F, T \otimes F)\}$. The composition is $F \circ G = \{(\beta\gamma) \cdot (a, c) : \beta \cdot (a, b) \in F, \gamma \cdot (b, c) \in G\}$ where $\alpha, \beta, \gamma \mathbb{Z}_{\geqslant 0}$.

This is a semiringed category with $S_{\mathcal{M}} = \mathbb{N}$ in which the counting problem is $\#\mathsf{P}$-hard, and the familiar category of counting constraint satisfaction problems can be expressed as a symmetric monoidal subcategory. Every constraint satisfaction problem can be visualized by a bipartite graph. One independent set is the variables,

each of which can be assigned the value true or false. Thus we allow the morphisms in $\mathrm{Hom}(\mathbb{1}, B^{\otimes n})$ for each n of the form $\{1 \cdot (\mathbb{1}, F \otimes F \otimes \cdots F), 1 \cdot (\mathbb{1}, T \otimes T \otimes \cdots T)\}$. The other independent set is the constraints. So we allow only those morphisms in $\mathrm{Hom}(B^{\otimes n}, \mathbb{1})$ whose coefficients are only zero or one. The string diagrams in this category will then be precisely the graphs visualizing boolean constraint satisfaction problems. In the sequel we generally use the set $\{0,1\}$ instead of $\{T, F\}$ to conform with common usage.

A strict monoidal functor $F : \mathcal{M} \to \mathcal{M}'$ between semiringed categories is *count preserving* if the induced map $F : S_{\mathcal{M}} \to S_{\mathcal{M}'}$ is a injective morphism of semirings. Schemes that generalize holographic algorithms [19] seek a count-preserving functor from an category with a FP counting problem to a category with an (in general) $\#\mathsf{P}$-hard counting problem.

In each type of circuit, we consider two semiringed categories $\mathcal{C}$ and $\mathcal{S}$. Let $\mathcal{L}$ be a problem of interest. We call $\mathcal{C}$ the *counting* category and $\mathcal{S}$ is a subcategory of $\mathrm{Vect}_{\mathbb{C}}$. Then let $f : \mathcal{L} \to \mathcal{C}$ be a map that gives an interpretation or encoding of the problem as a string diagram in $\mathcal{C}$. By this we mean that for every instance of a problem $l \in \mathcal{L}$, $f(l)$ is a string diagram that solves this instance of the problem.

The category $\mathcal{C}$ may have a non-intuitive encoding of the problem but has the advantage that there exists a polynomial-time algorithm to determine which morphism of $S_{\mathcal{M}} = \mathrm{Hom}(\mathbb{1}, \mathbb{1})$ is represented by an arbitrary monoidal word. We also have an interpretation $g : \mathcal{L} \to \mathcal{S}$. Then we want a monoidal functor h such that the diagram

$$\begin{tikzcd} \mathcal{L} \arrow[r, "f"] \arrow[dr, "g"'] & \mathcal{C} \arrow[d, "h"] \\ & \mathcal{S} \end{tikzcd}$$

commutes and such that the count is preserved by h. $\mathcal{S}$ is the subcategory generated by the morphisms in the image of either $h \circ f$ or g. The induced maps on $S_{\mathcal{C}}$ and $S_{\mathcal{S}}$ makes $S_{\mathcal{C}}$ a sub-semiring of $S_{\mathcal{S}}$. The functor h is called sDet, and sPf for determinantal and Pfaffian circuits respectively in the sequel.

Of course, it is important that the construction represented by the functors is implementable in polynomial time. Often this is not a concern, because diagrams in $\mathcal{C}$ and $\mathcal{L}$ are effectively identified, and the problem is expressed in the language that will be used to perform the contraction.

3. Determinantal circuits

Suppose X is a $n \times m$ matrix of elements of a field k with rows and columns labeled by finite disjoint subsets N and M of $\mathbb{N} = \mathbb{Z}_{\geqslant 0}$. For $i \in \mathbb{N}$, let $V_i \cong k^2$ be spanned by an orthonormal basis (with inner product) $v_{i,0}, v_{i,1}$ and for finite $N \subset \mathbb{N}$ write $V_N := \otimes_{i \in N} V_i$. Define the function sDet by

$$\mathrm{sDet} : \mathrm{Mat}_k(n, m) \to V_N^* \otimes V_M \cong (\mathbb{C}^{2*})^{\otimes n} \otimes (\mathbb{C}^2)^{\otimes m}$$

$$\mathrm{sDet}(X) = \sum_{I \subset [n], J \subset [m]} \det(X_{IJ}) |I\rangle\langle J|$$

where $|I\rangle = \bigotimes_{i \in N} v_{i, \chi(i,I)}$, $\langle J| = \bigotimes_{i \in M} v^*_{i, \chi(i,J)}$ and the indicator function $\chi(i, I) = 0$ if $i \notin I$ and 1 if $i \in I$.

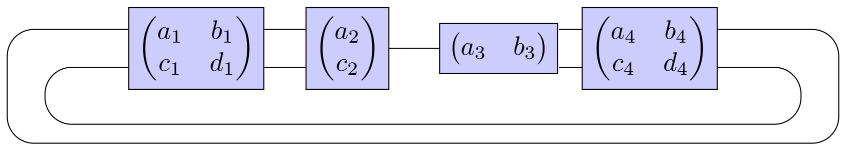


FIGURE 1. An example of a determinantal circuit (wires oriented clockwise). The four tensors in $\mathrm{Vect}_{\mathbb{C}}$, from left to right, are obtained by applying sDet to each matrix. Letting $V = \mathbb{C}^2$, they lie in $(V^*)^{\otimes 2} \otimes V^{\otimes 2}$, $(V^*)^{\otimes 2} \otimes V$, $V^* \otimes V^{\otimes 2}$, and $(V^*)^{\otimes 2} \otimes V^{\otimes 2}$ respectively.

This subdeterminant function sDet induces a strong monoidal functor sDet : $\mathcal{C} \to \mathrm{Vect}_{\mathbb{C}}$ from a matrix category to a subcategory $\mathcal{D}$ of $\mathrm{Vect}_{\mathbb{C}}$. Let $\mathcal{C}$ be the free monoidal category described as follows. The objects of $\mathcal{C}$ are finite ordered subsets of $\mathbb{N}$ (which may have repeated elements), with monoidal product union. The morphisms are $\mathbb{C}$-valued matrices with rows and columns labeled by subsets of $\mathbb{N}$. If M, N are two matrices with the set of row labels of M equal to the set of column labels of N, order them and let $N \circ M = NM$ be the ordinary matrix product, with the resulting matrix inheriting the row labels of N and the column labels of M. The monoidal product $\otimes_{\mathcal{C}}$ is the direct sum of labeled matrices.

Let $\mathcal{D}$ be the image of $\mathcal{C}$ in $\mathrm{Vect}_{\mathbb{C}}$. It will be free dagger symmetric traced [9] monoidal subcategory of finite-dimensional $\mathbb{C}$-vector spaces generated by the object $\mathbb{C}^2$, endowed with an orthonormal basis, and morphisms $\mathrm{sDet}(M)$ for M a labeled matrix. Tensor product and composition/contraction are the usual operations.

For $f : A \to A \in \mathcal{C}$, define $\mathrm{tr}(f) = \det(I + f)$ and define trace in $\mathcal{D}$ in the usual way. For a matrix M representing f, $\mathrm{tr}(M) = \sum_{I \subset [|M|]} \det M_I$.

Proposition 3.1. *$\mathcal{C}$ is a strict dagger symmetric monoidal category.*

Proof. We have already defined $\otimes$ for $\mathcal{C}$. We need to show that it is a bifunctor. For $A \subset \mathbb{N}$, id_A is the identity matrix with row and column labels A. It is easy to see that for any $A, B \subset \mathbb{N}$, $\mathrm{id}_A \otimes \mathrm{id}_B = \mathrm{id}_{A \otimes B}$. Now for morphisms $W, X, Y, Z \in \mathrm{Mor}(\mathcal{C})$, $W \otimes X \circ Y \otimes Z = (W \oplus X)(Y \oplus Z) = WY \oplus XZ = (W \circ Y) \otimes (X \circ Z)$, so $\otimes$ is indeed a bifunctor.

For $A, B, C \in \mathrm{Ob}(\mathcal{C})$, the associator $\alpha_{ABC} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$ is just equality by the associativity of matrix direct product. The unit for $\mathcal{C}$, denoted $\mathbb{1}$, is the empty set. Then $\lambda_A : \mathbb{1} \otimes A \to A$ and $\rho_A : A \otimes \mathbb{1} \to A$ are also equality since it is union with $\varnothing$. It is clear that α, λ, and ρ are natural isomorphisms.

We need to check that the diagrams from MacLane's Coherence Theorem commute. First let us check, for $A, B \in \mathrm{Ob}(\mathcal{C})$:

$$\begin{tikzcd} (A \otimes \mathbb{1}) \otimes B \arrow[r, "\alpha"] \arrow[dr, "\rho_A \otimes \mathrm{id}_B"'] & A \otimes (\mathbb{1} \otimes B) \arrow[d, "\mathrm{id}_A \otimes \lambda_B"] \\ & A \otimes B \end{tikzcd}$$

$(A \otimes \mathbb{1}) \otimes B = (A \cup \varnothing) \cup B$ is mapped to $A \cup B$ by $\rho_A \otimes \mathrm{id}_B$ via equality. Then α maps $(A \cup \varnothing) \cup B$ to $A \cup (\varnothing \cup B)$ via equality. This is then mapped to $A \cup B$ by $\mathrm{id}_A \otimes \lambda_B$ via equality, and the diagram commutes.

Now let us check the second diagram, for $A, B, C, D \in \mathrm{Ob}(\mathcal{C})$:

$$\begin{tikzcd}
& ((C \otimes A) \otimes B) \otimes D \arrow[r, "\alpha\otimes\mathrm{id}_D"'] \arrow[dl, "\alpha"] & (C \otimes (A \otimes B)) \otimes D \arrow[dd, "\alpha"] \\
(C \otimes A) \otimes (B \otimes D) \arrow[dr, "\alpha"] & & \\
& C \otimes (A \otimes (B \otimes D)) & C \otimes ((A \otimes B) \otimes D) \arrow[l, "\mathrm{id}_C \otimes\alpha"]
\end{tikzcd}$$

The object $((C \otimes A) \otimes B) \otimes D) = ((C \cup A) \cup B) \cup D$ is mapped to $C \cup (A \cup (B \cup D))$ by $(\mathrm{id}_C \otimes \alpha) \circ (\alpha) \circ (\alpha \otimes \mathrm{id}_D)$ via equality. Similarly, it is mapped to $C \cup (A \cup (B \cup D))$ by $\alpha \circ \alpha$ via equality. This diagram also commutes and so $\mathcal{C}$ is a monoidal category. Furthermore, since α, λ, and ρ are equalities, $\mathcal{C}$ is a strict monoidal category.

The braiding for $\mathcal{C}$ is a map $c_{A,B} : A \otimes B \to B \otimes A$, $A, B \in \mathrm{Ob}(\mathcal{C})$. It is given by the matrix

$$c_{A,B} = \begin{array}{c c} & \begin{array}{cc} B & A \end{array} \\ \begin{array}{c} A \\ B \end{array} & \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \end{array}$$

We need to check that the following diagrams commute for $A, B, C \in \mathrm{Ob}(\mathcal{C})$:

$$\begin{tikzcd}
& (B \otimes A) \otimes C \arrow[r, "\alpha"] & B \otimes (A \otimes C) \arrow[dr, "\mathrm{id}_B \otimes c_{A,C}"] & \\
(A \otimes B) \otimes C \arrow[ur, "c_{A,B}\otimes\mathrm{id}_C"] \arrow[dr, "\alpha"'] & & & B \otimes (C \otimes A) \\
& A \otimes (B \otimes C) \arrow[r, "c_{A,(B\otimes C)}"] & (B \otimes C) \otimes A \arrow[ur, "\alpha"'] &
\end{tikzcd}$$

$$\begin{tikzcd}
& (B \otimes A) \otimes C \arrow[r, "\alpha"] & B \otimes (A \otimes C) \arrow[dr, "\mathrm{id}_B \otimes c^{-1}_{A,C}"] & \\
(A \otimes B) \otimes C \arrow[ur, "c^{-1}_{A,B}\otimes\mathrm{id}_C"] \arrow[dr, "\alpha"'] & & & B \otimes (C \otimes A) \\
& A \otimes (B \otimes C) \arrow[r, "c^{-1}_{A,(B\otimes C)}"] & (B \otimes C) \otimes A \arrow[ur, "\alpha"'] &
\end{tikzcd}$$

The first diagram commutes by noting that

$$\begin{array}{cc} & \begin{array}{ccc} B & A & C \end{array} \\ \begin{array}{c} A \\ B \\ C \end{array} & \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{array}
\begin{array}{cc} & \begin{array}{ccc} B & A & C \end{array} \\ \begin{array}{c} B \\ A \\ C \end{array} & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{array}
\begin{array}{cc} & \begin{array}{ccc} B & C & A \end{array} \\ \begin{array}{c} B \\ A \\ C \end{array} & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \end{array} =$$

$$\begin{array}{cc} & \begin{array}{ccc} A & B & C \end{array} \\ \begin{array}{c} A \\ B \\ C \end{array} & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{array}
\begin{array}{cc} & \begin{array}{ccc} B & C & A \end{array} \\ \begin{array}{c} A \\ B \\ C \end{array} & \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \end{array}
\begin{array}{cc} & \begin{array}{ccc} B & C & A \end{array} \\ \begin{array}{c} B \\ C \\ A \end{array} & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{array}$$

The second diagram commutes since $c_{B,A}^{-1} = c_{A,B}$ (which implies the category is symmetric) for any $A, B \in \mathrm{Ob}(\mathcal{C})$ and so the second diagram is the same as the first.

The dagger for $\mathcal{C}$ is given by matrix transpose and the identity on objects. Clearly $\mathrm{id}_A^\dagger = \mathrm{id}_A^T = \mathrm{id}_A$. Given $X, Y \in \mathrm{Mor}(\mathcal{C})$, $X : A \to B$, $Y : B \to C$, $(X \circ Y)^\dagger = (XY)^T = Y^T X^T = Y^\dagger \circ X^\dagger : C \to A$. Lastly $X^{\dagger\dagger} = X^{TT} = X$.

We also need the dagger to satisfy two extra properties since we are working in a monoidal category. First, given $X, Y \in \mathrm{Mor}(\mathcal{C})$, $(X \otimes Y)^\dagger = (X \oplus Y)^T = X^T \oplus Y^T = X^\dagger \otimes Y^\dagger$. Secondly, α, λ, and ρ should all be unitary (its inverse is equal to its dagger). Since they are all the identity morphism, this is also satisfied. Thus $\mathcal{C}$ is indeed a strict dagger symmetric monoidal category. □

Theorem 3.2. *The map* sDet *defines a strict monoidal functor which is an equivalence (in fact, an isomorphism) of dagger symmetric traced categories. Thus while computing a trace in* $\mathrm{Vect}_{\mathbb{C}}$ *is in general* $\#P$*-hard, in the image of* sDet *it can be computed in polynomial time.*

We prove this in two parts as Lemmata 3.3 and 3.4.

Lemma 3.3. *The map* sDet *defines a strict monoidal functor which is an equivalence of monoidal categories.*

Proof. We have defined how sDet acts on matrices, and so morphisms in $\mathcal{C}$. It takes an object $A \in \mathrm{Ob}(\mathcal{C})$ to $\mathrm{sDet}(A) = V_A = \otimes_{i \in A} V_i$.

First we must show that sDet is a functor, i.e. that it respects composition and that $\mathrm{sDet}(\mathrm{id}_A) = \mathrm{id}_{\mathrm{sDet}(A)}$. Suppose $X \in \mathrm{Hom}_{\mathcal{C}}(I, J)$, $Y \in \mathrm{Hom}_{\mathcal{C}}(J, K)$ so X is a matrix with row labels I, column labels J and Y has row labels J and column labels K:

$$\mathrm{sDet}(Y) \circ \mathrm{sDet}(X) = \sum_{i \subseteq I} \sum_{j \subseteq J} \sum_{k \subseteq K} \det(X_{ij}) \det(Y_{jk}) |i\rangle\langle k|$$

$$= \sum_{i \subseteq I} \sum_{j \subseteq J} \det(XY_{ik}) |i\rangle\langle k| = \mathrm{sDet}(XY)$$

where the middle equality is the Cauchy-Binet formula. Now in $\mathcal{C}$, id_A is the identity matrix with row and column labels A. Then $\mathrm{sDet}(\mathrm{id}_A) = \sum_{I \subseteq A} |I\rangle\langle I|$ which is the identity morphism for the object $\mathrm{sDet}(A)$ in $\mathcal{D}$, and sDet is indeed a functor.

For sDet to be a monoidal functor, we must demonstrate two additional properties. First we must show that $\mathrm{sDet}(A \oplus B) = \mathrm{sDet}(A) \otimes \mathrm{sDet}(B)$. Let I and J be the rows and columns of A, respectively. Let I' and J' be likewise for B. A straightforward calculation gives

$$\mathrm{sDet}(A \oplus B) = \sum_{U \subseteq I \cup I'} \sum_{V \subseteq J \cup J'} \det(A \oplus B)_{UV} |U\rangle\langle V|$$

$$= \sum_{U \subseteq I \cup I'} \sum_{V \subseteq J \cup J'} \det(A_{U \cap I, V \cap J}) \det(B_{V \cap I', V \cap J'}) |U \cap I\rangle |U \cap I'\rangle\langle V \cap J| \langle V \cap J'|$$

$$= \sum_{U \subseteq I} \sum_{U' \subseteq I'} \sum_{V \subseteq J} \sum_{V' \subseteq J'} \det(A_{UV}) \det(B_{U'V'}) |U\rangle |U'\rangle\langle V| \langle V'| = \mathrm{sDet}(A) \otimes \mathrm{sDet}(B).$$

Secondly we must show there are morphisms $F_0 : \mathbb{1}_{\mathcal{D}} \to \mathrm{sDet}(\mathbb{1}_{\mathcal{D}})$ (the unit in $\mathcal{D}$ is the base field $\mathbb{C}$) and for any $A, B \in \mathrm{Ob}(\mathcal{C})$, $F_1 : \mathrm{sDet}(A) \otimes \mathrm{sDet}(B) \to \mathrm{sDet}(A \otimes B)$ satisfying certain axioms expressed as commutative diagrams.

Since $\mathrm{sDet}(\varnothing) = \otimes_{i\in\varnothing} V_i = \mathbb{C}$, F_0 is simply equality. Similarly for objects A and B,

$\mathrm{sDet}(A \otimes B) = \mathrm{sDet}(A \cup B) = \otimes_{i\in A\cup B} V_i = (\otimes_{i\in A} V_i) \otimes (\otimes_{j\in B} V_j) = \mathrm{sDet}(A) \otimes \mathrm{sDet}(B)$; so F_1 is equality. In the following diagrams, we shall call sDet simply F.

Let α', λ', ρ' be the natural transformations for $\mathcal{D}$. Note that all three are equalities. For $A, B, C \in \mathrm{Ob}(\mathcal{C})$, the following must commute:

$$\begin{tikzcd}
F(A) \otimes (F(B) \otimes F(C)) \arrow[r, "\alpha'"] \arrow[d, "\mathrm{id}_{F(A)} \otimes F_1"'] & (F(A) \otimes F(B)) \otimes F(C) \arrow[d, "F_1 \otimes \mathrm{id}_{F(C)}"'] \\
F(A) \otimes (F(B \otimes C)) \arrow[d, "F_1"'] & (F(A \otimes B) \otimes F(C)) \arrow[d, "F_1"'] \\
F(A \otimes (B \otimes C)) \arrow[r, "F(\alpha)"] & F((A \otimes B) \otimes C)
\end{tikzcd}$$

$$\begin{tikzcd}
F(B) \otimes \mathbb{1}' \arrow[r, "\rho'"] \arrow[d, "\mathrm{id}_{F(B)} \otimes F_0"] & F(B) \\
F(B) \otimes F(\mathbb{1}) \arrow[r, "F_1"] & F(B \otimes \mathbb{1}) \arrow[u, "F(\rho)"']
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathbb{1}' \otimes F(B) \arrow[r, "\lambda"] \arrow[d, "F_0 \otimes \mathrm{id}_{F(B)}"'] & F(B) \\
F(\mathbb{1}) \otimes F(B) \arrow[r, "F_1"] & F(\mathbb{1} \otimes B) \arrow[u, "F(\lambda)"']
\end{tikzcd}$$

The diagrams trivially commute as all of the maps are identities. So sDet is a strong monoidal functor. Since F_0, F_1 are equalities, it is in a fact a strict monoidal functor.

Lastly, we want to say that $\mathcal{C}$ and $\mathcal{D}$ are equivalent as monoidal categories. By definition of $\mathcal{D}$, sDet surjects onto objects and morphisms, so it is a full functor. Now consider $\mathrm{Hom}(A, B)$ for objects $A, B \in \mathrm{Ob}(\mathcal{C})$. Let $X \in \mathrm{Hom}(A, B)$. $\mathrm{sDet}(X)$ contains all the entries of X as coefficients in the sum since the entries of X are 1×1 minors, and X is determined by its image $\mathrm{sDet}(X)$. Thus sDet induces an injection on $\mathrm{Hom}(A, B) \to \mathrm{Hom}(\mathrm{sDet}(A), \mathrm{sDet}(B))$, and the functor is faithful. Thus it is an equivalence. However, it is not quite an isomorphism as sDet does not give a bijection on objects. $\square$

We have yet to define the braiding and dagger for $\mathcal{D}$ required to state Theorem 3.2. For sDet to respect the braiding, we need the following diagram to commute:

$$\begin{tikzcd}
F(A) \otimes F(B) \arrow[r, "F_1"] \arrow[d, "c_{F(A),F(B)}"'] & F(A \otimes B) \arrow[d, "F(c_{A,B})"] \\
F(B) \otimes F(A) \arrow[r, "F_1"] & F(B \otimes A)
\end{tikzcd}$$

Since F_1 is the identity, we just need that $c_{F(A),F(B)} = F(c_{A,B})$. $F(c_{A,B}) = |00\rangle\langle 00| + |01\rangle\langle 10| + |10\rangle\langle 01| - |11\rangle\langle 11|$. We define the braiding for $\mathcal{D}$ to be this. For the dagger, consider $X \in \mathrm{Mor}(\mathcal{C})$ with row labels I and column labels J.

$$\mathrm{sDet}(X^\dagger) = \sum_{i\subseteq I, j\subseteq J} \det(X_{ij}^T)|i\rangle\langle j| =$$

$$\sum_{i\subseteq I, j\subseteq J} \det(X_{ij})|j\rangle\langle i| = \mathrm{sDet}(X)^T$$

So the dagger for $\mathcal{D}$ is the normal dagger in Vect_k.

Again for $f : A \to A \in \mathcal{C}$, define $\mathrm{tr}(f) = \det(I + f)$ and define trace in $\mathcal{D}$ in the usual way.

Lemma 3.4. *The map* sDet *defines a strict monoidal functor which is an equivalence of dagger symmetric traced categories.*

Proof. By construction, sDet respects the braiding. We also showed that this functor respects the normal dagger for linear transformations. Theorem 3.6 and Proposition 3.7 below shows that sDet induces the identity map from $\mathrm{Hom}(\mathbb{1}_{\mathcal{C}}, \mathbb{1}_{\mathcal{C}}) \to \mathrm{Hom}(\mathbb{1}_{\mathcal{D}}, \mathbb{1}_{\mathcal{D}})$ and thus respects the trace.

$\square$

Remark 3.5. *This braiding is* not *the usual braiding for* $\mathrm{Vect}_{\mathbb{C}}$*. Thus while the functor* sDet *is count-preserving, the count will not be the same as if the standard braiding is used.*

Using the operations of $\oplus$ and matrix multiplication, we can transform any string diagram in $\mathcal{C}$ into a diagram with a single matrix, M, and thus evaluate the determinantal circuit efficiently.

Theorem 3.6. *The time complexity of computing the trace of a determinantal circuit in $\mathcal{C}$ is $O(dw^\omega) = O(dw^\omega + c^\omega)$ where d is the depth of the circuit, w is the maximum width, c is width at the input and output (so can be chosen to be the minimum width), and ω is the exponent of matrix multiplication.*

Proof. We have an $n \times n$ matrix with equal row and column labels, which we may assume to be $1, \ldots, n$. Then

$$\mathrm{sDet}(M) = \sum_{I,J \subseteq [n]} \det(M_{I,J}) |I\rangle\langle J|$$

and contracting this against itself gives

$$\sum_{I,J \subseteq [n]} \det(M_{I,J}) \langle J|I\rangle\langle J|I\rangle = \sum_{I \subset [n]} \det M_{I,I}.$$

That is, the trace of a matrix M in $\mathcal{C}$ is the exponentially large sum of its 2^n principal minors; we claim that $\det(I + A)$ is precisely this sum (Proposition 3.7). This is a classically known fact but we include a proof here for completeness. This enables us to compute this number in time n^ω. $\square$

The following identity is well-known (e.g. it can be derived from results in [7]); we include a proof for completeness.

Proposition 3.7. *Given two $n \times n$ matrices N, M, with N diagonal,*

$$\det(N + M) = \sum_{I \subseteq [n]} \det(M_I) \det(N_{\bar{I}})$$

where $M_I = M_{I,I}$ and $N_{\bar{I}}$ is the minor of N formed by removing the rows and columns labeled I. In particular, $\det(I + M) = \sum_{I \subseteq [n]} \det(M_I)$*.*

Proof. Let u_i be the columns of M and v_i the columns of N, $i = 1, \ldots, n$. Then $\det(N + M) = \bigwedge_{i=1}^n (u_i + v_i)$. Now let $\beta_1, \ldots, \beta_n$ be variables and $\gamma : \beta_i \mapsto \{v_i, u_i\}$ be an assignment of either v_i or u_i to every variable β_i. There are of course 2^n such

assignments. Summing over all possible γ, we can expand the above into the sum of "simple wedges":

$$\det(N+M) = \sum_\gamma \bigwedge_{i=1}^{n} \gamma(\beta_i)$$

Let us fix a particular γ. Let W be the matrix whose i^{th} column is $\gamma(\beta_i)$. Then

$$\det(W) = \bigwedge_{i=1}^{n} \gamma(\beta_i)$$

Let $U = \{i : \beta_i = u_i\}$ ordered in increasing order. Let $\bar{U} = \{i : \beta_i = v_i\}$, also ordered in increasing order. Then $W_U = M_U$ and $W_{\bar{U}} = N_{\bar{U}}$.

Now we must show that $\det(W) = \det(M_U)\det(N_{\bar{U}})$. Since U is an ordered set, there is no ambiguity in letting $U(j)$ be the j^{th} entry of U. Consider the matrix transformation where the element $W_{U(j),U(j)} \mapsto W_{j,j}$. This is done by swapping rows j and $U(j)$ and then swapping columns j and $U(j)$. Since we do two swaps, this transformation does not change the determinant; do this for all $j \in [|U|]$. Then perform the matrix transformation $W_{\bar{U}(j),\bar{U}(j)} \mapsto W_{j,j}$ for all $j \in [|\bar{U}|]$. This yields the following matrix:

$$\begin{bmatrix} M_U & 0 \\ X & N_{\bar{U}} \end{bmatrix}$$

Here X is the portion of columns of M specified by U that don't lie in the minor M_U (under the above permutation). Since N is diagonal, the entries of the columns of N specified by $\bar{U}$ not in the minor $N_{\bar{U}}$ will always be 0. The determinant of this matrix is $\det(M_U)\det(N_{\bar{U}})$,. Summing over all of assignments γ yields the result. □

Note that while $\mathcal{D}$ could be equipped with the object duality structure (A, A^*, i_A, e_A) from the category of finite-dimensional vector spaces to obtain a dagger closed compact category, the matrix category $\mathcal{C}$ is *not* a closed compact category: it lacks the morphisms i_A and e_A. The morphism $e_A : A \otimes A^* \to I$ would have to be the sDet of a 2×0 matrix, or the composition of several morphisms to obtain one of this type.

Proposition 3.8. *$\mathcal{C}$ does not have duals for objects.*

Proof. We cannot have $e_A = \mathrm{sDet}(M)$ for any M. The morphism we want is $|00\rangle + |11\rangle$, but there is a unique 2×0 matrix M and $\mathrm{sDet}(M) = |00\rangle$.

□

As a consequence, we really do have to work with traced categories [9] rather than the more convenient dagger closed compact categories.

A diagram in the equivalent categories $\mathcal{C}, \mathcal{D}$ is called a *determinantal circuit*; when the morphism represented is a field element, it computes the partition function, i.e. counts the weighted number of solutions to the $\#CSP$ it represents. Because these categories have a traced, dagger braided monoidal category structure, they come with a corresponding graphical language [18].

It is also a question of interest which tensors are determinantal. One can test whether a vector can be the set of determinants of minors from a matrix using the Plücker relations to obtain the relations among general minors of matrices. On the other hand, for minors of a fixed size this is an open problem [2].

4. Applications

4.1. **Multipaths.** We now discuss a specific diagrammatic language, and a way of viewing what it is a determinantal circuit counts in terms of *multipaths*. This point of view makes it easier to apply determinantal circuits to specific counting problems.

Suppose we have a determinantal circuit. Our convention shall be that objects will be represented by arrows moving from left to right. Vertical stacking will denote tensor product, ordered from top to bottom. In our circuit, we can group the morphisms into a sequence of *stacks*, where each stack is simply a group of morphisms which are aligned vertically. Each stack is also a single morphism obtained by tensoring the morphisms in the stack.

Consider the k^{th} stack in a determinantal circuit and suppose the incoming wires are labeled with the set I_k. For $i_k \subseteq I_k$, let χ_{i_k} be the binary string $b_{I_{k,1}} b_{I_{k,2}} \cdots b_{I_{k,y}}$ where $I_{k,m}$ is the m^{th} element of I_k (ordered from top to bottom in the diagram), $y = |I_k|$, and where $b_{I_{k,m}} = 1$ if $I_{k,m} \in i_k$ and 0 otherwise. The outgoing wires are labeled with the set I_{k+1} and for $i_{k+1} \subseteq I_{k+1}$. The k^{th} stack can be written as a morphism in $\mathcal{D}$ as

$$\sum_{i_k, i_{k+1}} W_{i_k, i_{k+1}} |\chi_{i_k}\rangle\langle\chi_{i_{k+1}}|$$

where $W_{i_k, i_{k+1}}$ is the appropriate scalar.

Now, for a fixed $i_k \subseteq I_k$, $i_{k+1} \subseteq I_{k+1}$, we say that $W_{i_k, i_{k+1}} |\chi_{i_k}\rangle\langle\chi_{i_{k+1}}|$ is a (weighted) multipath passing through the k^{th} stack. It specifies which incoming wires and outgoing wires of the k^{th} stack are activated. Note that unless $|i_k| = |i_{k+1}|$, $W_{i_k, i_{k+1}} = 0$ as it will be the determinant of a non-square matrix, justifying the term "multipath".

Now we say that a multipath through a determinantal circuit with n stacks is a tensor of the form

$$W_{i_1, i_2} W_{i_2, i_3} \cdots W_{i_n, i_1} |i_1\rangle\langle i_2|i_2\rangle\langle i_3| \cdots |i_n\rangle\langle i_1|$$

for fixed sets $i_1, i_2, \ldots, i_n$. If the determinantal circuit is collapsed to a matrix M, $\mathrm{sDet}(M)$ is the sum of tensors of the form

$$W_{i_1, i_2} \cdots W_{i'_n, i'_1} |i_1\rangle\langle i_2|i'_2\rangle \cdots \langle i_n|i'_n\rangle\langle i'_1|$$

However, unless $i_2 = i'_2, \ldots, i_n = i'_n$, this tensor will be zero. Also, given the cyclical invariance of the trace, it must be that $i_1 = i'_1$ as well. Thus the tensor represented by $\mathrm{sDet}(M)$ is the sum of weighted multipaths of the circuit. Taking the trace then sums up the weights.

Thus the value of a determinantal circuit is the sum of the weights of its multipaths. The definition we have given for multipath translates into a collection of paths, in the traditional sense, in the given determinantal circuit considered as a graph.

4.2. **Recovering the matrix tree theorem and variants.** We describe how to recover the Chung-Langlands rooted spanning forest theorem [5] by expressing it as a determinantal circuit. The theorem tells us that if B is the incidence matrix of a graph G with an arbitrary orientation placed on it, then the number of rooted spanning forests is $\det(I + BB^T)$. An example of a graph is given in Figure 2(a) and the determinantal circuit constructed for it in Figure 2(b). Our proof differs from that of Chung and Langlands in that we give an explicit interpretation of

every permutation in the calculation of the determinant of a principal minor as a structure in the graph.

Choose an arbitrary orientation on G, and consider a string diagram in $\mathcal{C}$ which is equipped with a $\mathcal{C}$-morphism (node) for every edge and vertex of G. A node corresponding to an edge is connected to a node corresponding to a vertex if the edge and vertex are incident in G. This edge is oriented toward the vertex node if that vertex is a sink for the edge in G; otherwise the edge is oriented towards the edge node. Note that this orientation is not the orientation reflecting primal and dual objects in the category. It is simply a tool to help explain the correct interpretation of the circuit. Let us call this portion of the diagram Z. It is portion of the circuit inside the dashed box in Figure 2(b).

To express the problem as a determinantal circuit, we will associate matrices (morphisms of $\mathcal{C}$) with the nodes of Z. Then applying the simplification as in the proof of Theorem 3.6 using the operations of $\oplus$ and matrix multiplication, we will obtain the matrix B. Let us imagine that the edge nodes of Z are arranged in a vertical fashion. To the right of them, we have the vertex nodes also arranged vertically. From top to bottom, vertex nodes will take values $\mathbf{1}, \mathbf{2}, \ldots$. Similarly, edges nodes will take values $\mathbf{a}, \mathbf{b}, \ldots$.

Now in Z, for every edge node, let there be a single edge extending to the left. Label this edge a if it comes from $\mathbf{a}$, b if it comes from $\mathbf{b}$, and so on. Next we label the edges from edge nodes to vertex nodes. Let the edges incident to $\mathbf{1}$ be labeled $1_1, 1_2, \ldots, 1_n$ where n is the degree of $\mathbf{1}$ in G. Do likewise for the node $\mathbf{2}$ and so on. Finally extend a single edge from every vertex node rightward.

For every edge node ϵ, associate a 1×2 matrix with it, M_ϵ. Let the entry in the first row and column of M_ϵ be -1 if the first column label denotes an output edge oriented away from ϵ. Otherwise, let it be 1. Define the other entry of M_ϵ similarly. So $M_\epsilon = [1 \ -1]$ or $[-1 \ 1]$.

For every vertex node ν, associate a $n \times 1$ matrix with it, M_ν. The row labels in order from top to bottom are $\nu_1, \nu_2, \ldots, \nu_n$ where n is the degree of ν in G. Let the output label be the number ν. Let M_ν have all entries 1.

Although in general we suppress it in pictures, whenever two wires cross, we put the braiding matrix (as discussed above) on the crossing.

Lemma 4.1. *Using the operations of matrix multiplication and $\oplus$, the matrices in Z collapse to the incidence matrix of G with some orientation placed on it.*

Proof. Let E be the matrix formed by taking the direct sum of all the matrices on the edge nodes. Let V be the direct sum of all the matrices on vertex nodes. $A = EV$ will be the matrix that the morphisms in Z collapse to.

The rows of E are labeled with the edges of G and the columns of V are labeled with the vertices of G. Let e be an edge of G and v a vertex. Let A_{ev} be the element of A in the e^{th} row and v^{th} column.

The e^{th} row of E contains a ± 1 in the column v_i and ± 1 in column v_j (where i and j are two distinct numbers between 1 and the degree of v). If the v^{th} column in V does not contain a 0 in both the v_i^{th} and v_j^{th} row (the case where e is not incident to v), then it will contain a 1 in either the v_i^{th} row or the v_j^{th} row, but not both as there are no two nodes in Z that have two edges in between them. In that case v is incident to e in G and A_{ev} is ± 1.

So A_{ev} is ± 1 if and only if e is incident to v in G and otherwise is 0. Now suppose that $A_{ev} = 1$. Let v' be the other endpoint of e. We need to say that $A_{ev'} = -1$.

Now let v_i be the edge in the diagram that connects the node corresponding to e to the node corresponding to v. Then the e^{th} row of E has a 1 in the v_i^{th} column. Then let v'_j be the edge in the diagram that connects the node corresponding to e to the node corresponding ton v'. Since the e^{th} row of E has exactly two nonzero entries, one being 1 and the other being -1, the $v'_j th$ column must have a -1 in the e^{th} row meaning that $A_{ev'} = -1$.

So indeed A is an incidence matrix of G with some orientation placed on it.

□

Now we reflect Z across a vertical line and take the transpose of all matrices associated with nodes of Z. We call this Z^T. Indeed, Z^T collapses to the matrix B^T. The outputs of Z are precisely the inputs of Z^T and vice versa. So we glue Z and Z^T together to get the circuit we want. In Figures 2(a) and (b), we give an example of a graph and its transformation into a circuit. Figure 2(c) is equivalent to 2(b) but we have multiplied our two stacks of vertex nodes.

Let us turn our attention to the kinds of multipaths our circuit admits. Consider $\mathrm{sDet}(M_\epsilon)$, where M_ϵ is the matrix in Z associated with the edge node ϵ. It is of the form $|0\rangle\langle 00| \pm |1\rangle\langle 10| \mp |1\rangle\langle 01|$. This says that for any multipath, ϵ cannot have two activated output edges. However, it may not have any path passing through it at all. The signs of the various tensors denote the weight of that part of the path.

Let M_ν be the matrix associated with the vertex node ν (vertex nodes as in the collapsed version in Figure 2(c)). Let n be the degree of the corresponding vertex ν in G. Since M_ν is a rank one matrix, the only nonzero principal minors are 1×1 minors and the empty matrix. So

$$\mathrm{sDet}(M_\nu) = |0_{\nu_1}\cdots 0_{\nu_n}\rangle\langle 0_{\nu_1}\cdots 0_{\nu_n}| + \sum_{i=1}^{n} |\chi_i\rangle\langle\chi_i|$$

where $|\chi_i\rangle = |0_{\nu_1}\cdots 1_{\nu_i}\cdots 0_{\nu_n}\rangle$ and $\langle\chi_i|$ defined similarly. This means that for any valid multipath, a vertex node either has exactly one activated input edge and one activated output edge or no path through it at all. It is not possible for two are more separate paths in the multipath to pass through ν.

Lemma 4.2. *Every multipath of ZZ^T is a subgraph of G.*

Proof. Here we are considering multipaths without their weights to show that every multipath of ZZ^T may be associated to a subgraph of G. For a particular multipath μ, we say that an edge node in Z is activated if there is a path in the multipath that passes through it. Similarly for edge nodes in Z^T. For this multipath, let $\mu(E)_Z$ be the set of activated edge nodes in Z and $\mu(E)_{Z^T}$ the activated edge nodes in Z^T. It must be that $\mu(E)_Z = \mu(E)_{Z^T}$. Then a multipath tells us which edges are activated in our subgraph, and we include those vertices incident to each of these edges.

□

In general, there may be more than one multipath that is associated with a particular subgraph of G. Furthermore, these different multipaths may have different weights.

First, let us consider the weight of a path in our multipath. If we consider our circuit to have three stacks (as in Figure 2(c)), let us consider the part of the path from the first stack to the second, p_1. If the subpath travels along an edge oriented

towards an edge node, the subpath has weight $+1$. If the edge is oriented towards a vertex node, the subpath has weight -1. This follows from the way we defined the matrix M_ϵ for an edge node ϵ. The subpath from the second stack to the third stack, p_2, satisfies the same properties.

Then the weight of the path is the weight of p_1 times the weight of p_2. So if a path has its two subpaths either both oriented towards a vertex node or both towards edge nodes, the path has weight $+1$. If one of the subpaths is oriented towards a vertex node and the other subpath is oriented towards an edge node, the weight of the path is -1. Now we say that the *unassociated weight* of a multipath is the product of the weights of its constituent paths.

Recall that a multipath μ must have $\mu(E)_Z = \mu(E)_{Z^T}$, so we can view μ as a permutation of the edges in G associated with the edge nodes in $\mu(E)_Z$. This is well defined as two different paths cannot emanate from the same edge node in Z and thus two different paths cannot both pass through the same edge node in Z^T.

Define the weight of a multipath to be its unassociated weight times the sign of this permutation. The reason for this is that every time two wires cross in the multipath, we put the braiding matrix at the crossing. This will flip the sign of the unassociated weight.

Definition 4.3. A *straightline* multipath is one were every path passes through e_Z and e_{Z^T} for some edge e. By e_Z, we mean the edge node associated with e in Z. e_{Z^T} is the edge node associated with e in Z^T.

Lemma 4.4. *Every straightline multipath has weight 1.*

Proof. Let μ be a straightline multipath. Viewing μ as a permutation, it is obviously the identity permutation so the weight of μ is the unassociated weight of μ. Now let us consider an arbitrary path, p, in μ. Let p connect the edge nodes e_Z and e_{Z^T} by passing through vertex node ν. Since ZZ^T is symmetric across a vertical line, the subpath from e_Z to ν and the subpath from ν to e_{Z^T} are either both oriented towards ν or both oriented towards their respective edge nodes. So this path has weight 1. Since the path we chose was arbitrary, multiplying all the constituent paths together gives that the weight of the multipath is 1.

□

Lemma 4.5. *Every spanning forest is represented only by straightline multipaths. Furthermore every spanning forest has a representation as a multipath.*

Proof. Since any spanning forest can be broken into the disjoint union of trees, it is sufficient to show trees only have straightline multipath representations. So let us suppose that there is a multipath μ that is not straightline that represents some tree. Since μ is not the identity permutation on the edges activated, we can write it as the product of disjoint cycles. Let us consider one of these cycles. This will be represented by a submultipath μ' with m paths. Then since μ' is cycle, without loss of generality,we can represent the paths P_i in μ' as $P_1 = e(1)_Z \to e(2)_{Z^T}$, $P_2 = e(2)_Z \to e(3)_{Z^T}, \cdots, P_m = e(m)_Z \to e(1)_{Z^T}$ However, this means that $e(1)$ and $e(2)$ share a vertex, $e(2)$ and $e(3)$ share a vertex, and so on. So μ' represents a cycle in G, and μ cannot represent a tree.

The second part of the lemma is obvious. In any spanning forest, there is a way to associate any edge with a vertex adjacent to it such that no two edges are associated with the same vertex. Then consider a straightline multipath where each path through an edge node travels through the associated vertex node.

□

Lemma 4.6. *If μ is a multipath which is a cyclic permutation, its weight is -1*

Proof. In the previous lemma, we showed that μ represents a cycle in G. Now suppose that the orientation we put on G was such that the arrows travel around the cycle. Let e and e' be two edges in this cycle that share a vertex v. In μ, there is either a path from e_Z to e'_{Z^T} or e'_Z to e_{Z^T}. Suppose the path is from e_Z to e'_{Z^T}. The subpath from e_Z to v in ZZ^T has the opposite weight of the subpath from v to e'_{Z^T}. So the path between the two has weight -1. If the path had been from e'_Z to e_{Z^T}, this would still be true.

Now suppose we flip an arrow on edge e. Then let e^- be one edge sharing a vertex with e and e^+ the other. Then the weight of the path from e^-_Z to e_{Z^T} (and from e_Z to $e^-_{Z^T}$) would flip sign. So would the weight of the path from e_Z to $e^+_{Z^T}$. So the unassociated weight of μ will remain the same. Since we can get to any orientation on this cycle by flipping arrows, the unassociated weight of this cycle is orientation independent.

Every path in μ has weight -1. So the unassociated weight of μ is $(-1)^k$, where k is the number of edges of the cycle. Now if k is odd, μ has unassociated weight -1. The sign of the permutation of μ is the sign of permuting an odd number of elements cyclically. The sign of such a permutation is 1. So the weight of μ is -1. If k is even, the unassociated weight of μ is 1. The sign of permuting an even number of elements cyclically is -1. So μ has weight -1 once again.

□

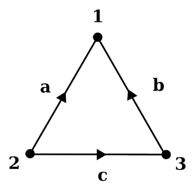
Thus we obtain our version of Theorem 2 of [5] as follows.

Theorem 4.7. *The value* $\operatorname{tr}(ZZ^T) = \det(I + B^TB)$ *is the number of rooted spanning forests of* G*. Furthermore,* $\det(Ix + BB^T)$ *gives a polynomial where the coefficient of* x^k *is the number of rooted spanning forests of* G *with* k *roots.*

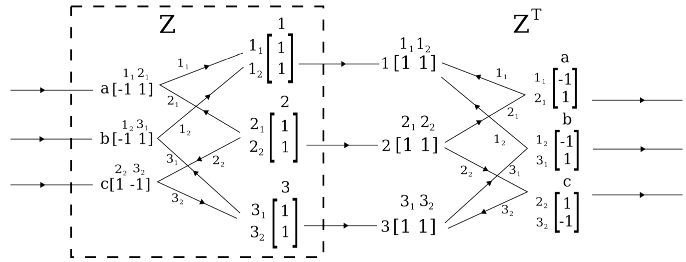
Proof. First let us consider a straightline multipath that represents a spanning forest in G. We look at the submultipaths representing the disjoint trees. If a multipath μ represents a tree with t edges, there are t paths in the μ. So one of the $t+1$ vertex nodes associated with a vertex in the tree does not have a path going through it. This will be the root of the tree.

Since μ is straightline, it has weight 1. So every multipath representing an rooted spanning forest contributes a +1 to the sum. It is not hard to see that there is exactly one straightline multipath for every rooted spanning forest and this is the only way to represent a particular rooted spanning forest making use of a previous lemma. We now need to argue that there are multiple ways to represent subgraphs with cycles and that their weights will cancel out.

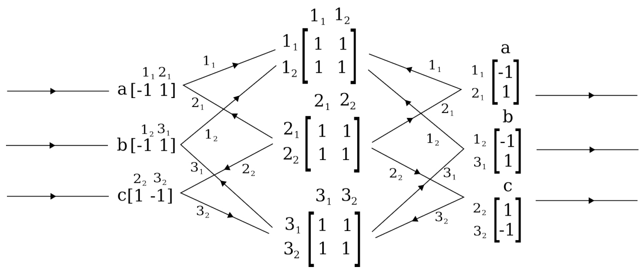
It turns out that every cycle has four multipath representations and that two have weight 1 and two have weight -1. This will be sufficient for proving the result. Let us consider a cycle in G with k edges $\{e(i)\}$. Then we can represent this cycle with paths $P_1 : e(1)_Z \to e(2)_{Z^T}, P_2 : e(2)_Z \to e(3)_{Z^T}, \ldots, P_k : e(k)_Z \to e(1)_{Z^T}$ which will have weight -1 as a multipath by the previous lemma. We can also represent it with $P'_1 : e(2)_Z \to e(1)_{Z^T}, P'_2 : e(3)_Z \to e(2)_{Z^T}, \ldots, P'_k : e(1)_Z \to e(k)_{Z^T}$ which is the previous multipath reflected across a vertical axis. This will also have weight -1.



(A) A rooted graph



(B) Its corresponding circuit



(C) Equivalent circuit

FIGURE 2. Transforming a rooted graph to a determinantal circuit.

These are the only two ways to represent a cycle in G where the permutation induced by the multipath is not the identity. This is because a nontrivial permutation tells us how to walk around a given subgraph and there are only two ways to walk around a cycle: clockwise and counter-clockwise. However, there will be two representations of a cycle with straightline multipaths. Consider what a straightline multipath is. It tells us which edges are activated and associates to each edge a vertex incident to it. In a cycle, there is only two ways to make such an identification.

Since we have two straightline multipaths representing the cycle, $+2$ is contributed to the overall value for every cycle. But of course this is canceled out the by the -2 contributed by the other two multipaths representing the cycle. Now if a multipath contains a submultipath that is a cycle, taking that submultipath and replacing it with one of opposite sign gives a multipath of the same subgraph with a different sign. So the only multipaths whose weights actually contribute to the sum are those representing rooted spanning forests.

In the above setup, the Laplacian would actually be $Z^T Z$. However, note that in the circuit, we can move the Z^T portion around so that the circuit has Z^T to the left of Z. This is to say that the trace of the category satisfies the trace property: $\det(I + XY) = \det(I + YX)$. The multipaths will be the same (albeit deformed) and will still retain the combinatorial meaning discussed above. When considering a minor of BB^T of size k, that corresponds to a multipath in $Z^T Z$ with k paths. We can think of this as activating k vertices in G.

We have already shown that $\det(I + B^T B) = \det(I + BB^T)$ is the total number of rooted spanning forests. Now consider

$$\det(Ix + BB^T) = \sum_{k=0}^{|V|} \Bigg(\sum_{\substack{I \subseteq [|V|] \\ |I| = |V|-k}} BB^T_I \Bigg) x^k$$

So a minor of BB^T of size $|V| - k$ is one where exactly k of the vertices don't have a path passing through them. By our correspondence, this means those k vertices are roots in the rooted spanning forest. So indeed this polynomial is a generating function for the number of rooted spanning forests where the exponent denotes the number of roots.

□

Now we discuss how to prove Kirchoff's Matrix Tree Theorem as a corollary. However, this is the statement about the cofactors of the Laplacian of a graph. So we once again consider the determinantal circuit $Z^T Z$ which gives the trace $\det(I + BB^T)$.

Suppose we have a determinantal circuit that collapses to a matrix M. Then take the determinant of some principal minor M'. If M' has labels I, this corresponds to the value of activating just the external edges (those that wrap around the circuit) labeled I in the circuit. Then taking $\det(I + M)$ is just summing up the values over all possible subsets of the external edges. Since the matrix tree theorem deals with cofactors, in our circuit $Z^T Z$, we will consider multipaths with all edges in the diagram turned on except for one.

Corollary 4.8. *(Kirchoff's Matrix Tree Theorem) The number of spanning trees of a graph is the absolute value of any cofactor of its Laplacian.*

Proof. Consider the absolute value of an arbitrary cofactor of $Z^T Z$. This will be the value of $Z^T Z$ will all the edges activated except for one. This corresponds to multipaths where all vertex nodes of our graph are activated except for one. Let the activated vertices be denoted by V'. We only need to consider straightline multipaths as all other multipaths will be canceled out.

Take a given straightline multipath. For every vertex node, there will be an associated edge node. So if $|V|$ is the number of vertices in our graph, we have an rooted spanning forest with $|V| - 1$ edges. This is a spanning tree. So the number of rooted spanning forests on V' is the number of spanning trees of our graph.

What is happening here is that we are specifying a root for our spanning trees (the vertex omitted). Which vertex we specify as the root doesn't matter. Then we are counting maximal spanning forests with that root. These will simply be spanning trees.

□

4.3. **Simulating quantum circuits in the presence of closed timelike curves.** Determinantal circuits define a class of tensor networks with a polynomial-time contraction algorithm. An immediate consequence is that certain types of quantum circuits (or more generally tensor networks possibly including preparations and postselection) can be simulated efficiently using this technique. Essentially these are the tensor networks of the type shown in Figure 1 (with arbitrarily many wires

and transformations): a block of multilinear logic with the same number of inputs and outputs, with those inputs and outputs connected via nested loops.

The loop in such a circuit corresponds to a postselected closed timelike curve (P-CTC) [13]. These are constructed from bell states (cups) and postselected bell costates (measurements that act as caps). The resulting logical category of circuits represent physical experiments (which, if they contain an embedded contradiction, have count zero [16]).

If postselection and preparation operations beyond those necessary to construct the cap and cup are not allowed, all the transformations represented by the boxes are restricted to be unitary. If sufficient additional postselection or preparation is allowed, arbitrary transformations of the form $\mathrm{sDet}(A)$ can be appear in the boxes. The resulting linear transformation can be interpreted physically in terms of a "quantum SVD" as in [16], where the amount of postselection or preparation required to represent an arbitrary linear transformation is also discussed.

5. Relation to Pfaffian circuits

Pfaffian Circuits [15] were introduced as a reformulation of matchcircuits [19]. We present an slightly different definition using category theory. We want to know what the relation of determinant circuits is with respect to Pfaffian circuits. The most natural way of doing this is functoriality.

We now define the category that gives us Pfaffian circuits. Consider the set $\mathscr{M} \times \{0,1\}$, where $\mathscr{M}$ is the set of labeled skew-symmetric matrices. Furthermore, the columns and rows should have the same labels in the same order. The label sets are subsets of $\mathbb{N}$. As before, for $i \in \mathbb{N}$, let $V_i \cong \mathbb{C}^2$ be spanned by an orthonormal basis (with inner product) $v_{i,0}, v_{i,1}$ and for $N \subset \mathbb{N}$ write $V_N := \otimes_{i\in N} V_i$. Now let us consider the following function:

$$\mathrm{sPf} : \mathscr{M} \times \{0,1\} \to V_N^* \otimes V_N$$

$$\mathrm{sPf}(M,0) = \sum_{I \subseteq N} \mathrm{Pf}(M_I)|I\rangle$$

$$\mathrm{sPf}(M,1) = \sum_{I \subseteq N} \mathrm{Pf}(M_{\bar{I}})\langle I|$$

where $|I\rangle = \bigotimes_{i\in N} v_{i,\chi(i,I)}$, $\langle J| = \bigotimes_{i\in M} v^*_{i,\chi(i,J)}$ and the indicator function $\chi(i,I) = 0$ if $i \notin I$ and 1 if $i \in I$. Here M_I means the principal minor of M with row and column labels I. $M_{\bar{I}}$ means the principal minor of M with the rows and columns labeled I removed. We will use the convention that $\mathrm{sPf}(M,0)$ will be denoted $\mathrm{sPf}(M)$ and $\mathrm{sPf}(M,1)$ will be denoted $\mathrm{sPf}^\vee(M)$.

The sPf function lets us define a monoidal subcategory of $\mathrm{Vect}_{\mathbb{C}}$. Let $\mathscr{P}$ be the free monoidal category defined as follows. The objects are of the form V_N for ordered subsets of $\mathbb{N}$, the tensor product being the usual one. The morphisms of $\mathscr{P}$ are generated by elements from the image of sPf. Composition and tensor product will be inherited from $\mathrm{Vect}_{\mathbb{C}}$.

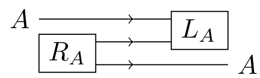
Theorem 5.1. *$\mathscr{P}$ is a strict monoidal category with daggers.*

Proof. By our definition of $\mathscr{P}$, it will be the smallest monoidal subcategory of $\mathrm{Vect}_{\mathbb{C}}$ containing the generating morphisms with the specified objects. Furthermore, we can assume we are dealing with the strict version of $\mathrm{Vect}_{\mathbb{C}}$ without loss of generality.

So the α, λ, and ρ maps that $\mathscr{P}$ inherits will be identities. We want to show that the identity morphism is actually generated by our specified morphisms. Consider the following matrix for an object A:

$$I_A = \begin{array}{c c} & \begin{array}{cc} A & A \end{array} \\ \begin{array}{c} A \\ A \end{array} & \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \end{array}$$

Let $L_A = \mathrm{sPf}(I_A) = |0_A 0_A\rangle + |1_A 1_A\rangle$ and $R_A = \mathrm{sPf}^\vee(I_A) = \langle 0_A 0_A| + \langle 1_A 1_A|$. Then we can contract these these two morphisms along a single edge as in the following picture:



This gives us the morphism $|0_A\rangle\langle 0_A| + |1_A\rangle\langle 1_A|$ which is the identity morphism on A. To prove that $\mathscr{P}$ has daggers, we need to prove a few other things first. □

Definition 5.2. The anti-transpose of a matrix N, denoted by $\hat{N}$, is N flipped across the non-standard diagonal.

Lemma 5.3. $\mathrm{Pf}(\hat{N}) = \mathrm{Pf}(N)$.

Proof. Let $N = \{\eta_{ij}\}$ be an $n \times n$ matrix. If n is odd, the above is trivial, so let n be even. Now let $\mathscr{F}$ be the set of partitions of $[n]$ into pairs, (i_k, j_k), $i_k < j_k$. If $\pi \in \mathscr{F}$ we can define the sign of π, $\mathrm{sgn}(\pi)$. This is done by considering the set $[n]$ as a sequence of nodes laid out horizontally and labeled $1, \ldots, n$ from left to right. Then if two nodes are paired in π, connect them with an edge. Then $\mathrm{sgn}(\pi)$ is $(-1)^k$ where k is the number of places where lines cross. Now we can define $\mathrm{Pf}(N)$ as follows:

$$\mathrm{Pf}(N) = \sum_{\pi \in \mathscr{F}} \mathrm{sgn}(\pi) \prod_{(i_k, j_k) \in \pi} \eta_{i_k j_k}$$

Now let $\eta'_{ij} = \eta_{n-j+1, n-i+1}$ be the entries of $\hat{N}$ and suppose $\pi \in \mathscr{F}$. Then the mapping $\mathscr{F} \to \mathscr{F} : \pi \mapsto \pi'$ given by $(i_k, j_k) \mapsto (n - j_k + 1, n - i_k + 1)$ is a bijective involution. Note that π' is the matching formed from π by relabeling the nodes as $n, \ldots, 1$ from left to right. This preserves the number of crossings of edges so that $\mathrm{sgn}(\pi') = \mathrm{sgn}(\pi)$. Thus we get

$$\mathrm{Pf}(\hat{N}) = \sum_{\pi \in \mathscr{F}} \mathrm{sgn}(\pi) \prod_{(i_k, j_k)} \eta'_{i_k j_k} =$$

$$\sum_{\pi' \in \mathscr{F}} \mathrm{sgn}(\pi') \prod_{(n-j_k+1, n-i_k+1)} \eta_{n-j_k+1, n-i_k+1} = \mathrm{Pf}(N).$$

□

Definition 5.4. If I is a bitstring. Let $\tilde{I}$ be the bitstring reflected across a vertical axis. Then $|\tilde{I}\rangle$ and $\langle\tilde{I}|$ are the tensors where the $\tilde{I}$ is formed from considering I as a bitstring. If $I \subseteq N$, $\tilde{I}$ is formed by considering I as bitstring representing a characteristic function. Then $\tilde{I}$ is a characteristic function defining another subset of N.

Corollary 5.5. *Let N be a skew symmetric matrix with labels M. Let $\hat{N}$ also have labels M.* $\mathrm{sPf}(\hat{N}) = \sum_{I \subseteq M} \mathrm{Pf}(N_I)|\tilde{I}\rangle$

Proof. Let $I \subseteq M$. Note that $N_I = \hat{N}_{\tilde{I}}$. Then $\mathrm{Pf}(N_I) = \mathrm{Pf}(\hat{N}_{\tilde{I}})$. This gives the result.

$\square$

Proposition 5.6. *For any skew-symmetric matrix M,*

$$\sum_I \mathrm{Pf}(M_I)\langle I|$$

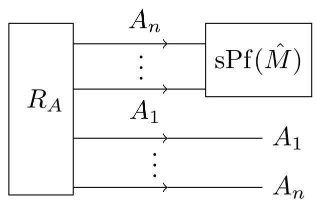
$$\sum_I \mathrm{Pf}(M_{\bar{I}})|I\rangle$$

are morphisms of $\mathscr{P}$. This implies that $\mathscr{P}$ is a dagger monoidal category.

Proof. Let M have labels $A = \{A_1, \dots, A_n\}$. Then $\hat{M}$ will have labels $\hat{A} = \{A_n, \dots, A_1\}$. Let R_A be defined as:

$$R_A = \mathrm{sPf}\ \begin{array}{c c} & \begin{array}{cc} \hat{A} & A \end{array} \\ \begin{array}{c} \hat{A} \\ A \end{array} & \begin{pmatrix} 0 & \tilde{I} \\ -\tilde{I}^T & 0 \end{pmatrix} \end{array}$$

where $\tilde{I}$ is the identity matrix reflected over a vertical axis. Then consider the following morphism in $\mathscr{P}$:



This diagram represents the morphism

$$\left(\sum_{I \subseteq A} \mathrm{Pf}(M_I)|\tilde{I}\rangle\right)\left(\sum_{I \subseteq \{\hat{A}A\}} \langle\tilde{I}|\langle I|\right) =$$

$$\sum_{I \subseteq A} \mathrm{Pf}(M_I)\langle I|$$

$\sum_I \mathrm{Pf}(M_{\bar{I}})|I\rangle$ can be formed similarly by instead using $\mathrm{sPf}^\vee(\hat{M})$ and $\mathrm{sPf}(R_A)$. Now since every generating morphism has a dagger, the entire category has a dagger and it is the usual vector space dagger. $\square$

Note that there are two primary types of morphisms in $\mathscr{P}$, namely those of the form $\mathrm{sPf}(M)$ and those of form $\mathrm{sPf}^\vee(M)$. The diagrams will then form bipartite graphs.

Suppose we are given a Pfaffian circuit Γ. Let Ξ_i be the gates of the form $\mathrm{sPf}(M)$ and Θ_i be the gates of the form $\mathrm{sPf}^\vee(M)$. We define $\Xi = \tilde{\bigoplus}_i \Xi_i$ and Θ likewise. $\tilde{\bigoplus}$ is the direct sum with the row and columns reordered by a particular ordering. The ordering is found by drawing a planar curve through the Pfaffian circuit such that every edge is intersected by the curve once and exactly once. Since a Pfaffian circuit is planar and bipartite, such a curve always exists and the curve chosen does not matter. The edges are then labeled based on when the curve intersects them. This is ordering used to define $\tilde{\bigoplus}$. $\breve{\Theta}$ is defined to be $\{(-1)^{i+j+1}\theta_{ij}\}$.

Theorem 5.7. *The value of a Pfaffian circuit* Γ *is given by* $\mathrm{Pf}(\Xi + \check{\Theta})$[15]

This tells us that Pfaffian circuits can be computed in polynomial time. Now we want to investigate if there is a functor that transforms determinantal circuits into Pfaffian circuits. We want the trace to be preserved so that the resulting Pfaffian circuit still solves the same problem as the original determinantal circuit. We also want the functor to be faithful.

Given a morphism in $\mathcal{D}$, we show how to construct it in $\mathscr{P}$. Now it is well know that for an $n \times n$ matrix M,

$$\mathrm{Pf}\begin{bmatrix} 0 & M \\ -M^T & 0 \end{bmatrix} = (-1)^{n(n-1)/2}\det(M)$$

We want a way to embed a matrix M into skew-symmetric matrix such that the Pfaffian corresponds to the determinant. Let us define $\tilde{M}$ as the matrix M reflected across a vertical axis. We define $S(M)$ to be

$$S(M) = \begin{bmatrix} 0 & \tilde{M} \\ -\tilde{M}^T & 0 \end{bmatrix}$$

Proposition 5.8. *For an* $n \times n$ *matrix* M*,*

$$\mathrm{Pf}(S(M)) = \mathrm{Pf}\begin{bmatrix} 0 & \tilde{M} \\ -\tilde{M}^T & 0 \end{bmatrix} = \det(M)$$

Proof. In general, $\tilde{M}$ can be made from M with $\lfloor \frac{n}{2} \rfloor$ column swaps. So if $n \equiv 0, 1$ modulo 4, $\lfloor \frac{n}{2} \rfloor$ is an even number and so $\det(\tilde{M}) = \det(M)$. Now if n is congruent to 0 or 1 modulo 4, then $\mathrm{Pf}(S(M)) = (-1)^{n(n-1)/2}\det(\tilde{M}) = \det(\tilde{M}) = \det(M)$. If n is congruent to 2 or 3 modulo 4, then $\lfloor \frac{n}{2} \rfloor$ is an odd number so $\det(\tilde{M}) = -\det(M)$ and $\mathrm{Pf}(S(M)) = (-1)^{n(n-1)/2}\det(\tilde{M}) = -\det(\tilde{M}) = \det(M)$.

□

Theorem 5.9. *Every morphism in* $\mathcal{D}$ *is a morphism in* $\mathscr{P}$*. Thus there is a trace-preserving faithful strict monoidal functor from* $\mathcal{D} \to \mathscr{P}$ *given by inclusion.*

Proof. First suppose that M is an $n \times n$ matrix. The labels of $S(M) = R \cup \tilde{C}$ where R is the row labels of M and C are the column labels of M. Now let K be a subset of the labels. Then let $I = K \cap R$ and $\tilde{J} = K \cap \tilde{C}$. Then we get

$$\mathrm{Pf}(S(M)_K) = \mathrm{Pf}\begin{bmatrix} 0 & \tilde{M}_{I,J} \\ -\tilde{M}_{I,J}^T & 0 \end{bmatrix} = \det(M_{I,J})$$

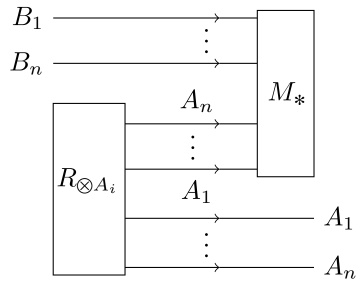
So we get that

$$\mathrm{sPf}(S(M)) = \sum_{I \subseteq R, J \subseteq C} \det(M_{I,J}|I\rangle|\tilde{J}\rangle$$
$$\mathrm{sPf}^\vee(S(M)) \sum_{I \subseteq R, J \subseteq C} \det(M_{\bar{I},\bar{J}})\langle I|\langle\tilde{J}|$$

Let us consider the identity morphism on $A_n \otimes \cdots \otimes A_1$ in $\mathcal{C}$. It is given by the matrix

$$I_{\otimes A_i} = \begin{array}{c} \\ A_n \\ A_{n-1} \\ \vdots \\ A_1 \end{array} \begin{array}{c} \begin{array}{cccc} A_n & A_{n-1} & \cdots & A_1 \end{array} \\ \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \end{array}$$

Suppose we have an $n \times n$ matrix $M : B_1 \otimes \cdots \otimes B_n \to A_1 \otimes \cdots \otimes A_n$. Then we define $M_* = \mathrm{sPf}(S(M))$ and $R_{\otimes A_i} = \mathrm{sPf}^\vee(S(I_{\otimes A_i}))$. Let us consider the morphism in $\mathscr{P}$ given by



For $I \subseteq \{B_1, \ldots, B_n\}$, $\tilde{J}, \tilde{J}' \subseteq \{A_n, \ldots, A_1\}$; and $J' \subseteq \{A_1, \ldots A_n\}$, we can represent this tensor as

$$\left(\sum \det(M_{I,J})|I\rangle|\tilde{J}\rangle\right)\left(\sum \langle\tilde{J}'|\langle J'|\right) =$$

$$\sum \det(M_{I,J})|I\rangle\langle J| = \mathrm{sDet}(M)$$

So for any square matrix M, $\mathrm{sDet}(M)$ is a morphism in $\mathscr{P}$. Now not every morphism in $\mathcal{C}$ is a square matrix. However, if we have an $n \times m$ matrix M, we can make it square. If $n < m$, then let $M' = M \oplus Z_{m-n}$ where Z_{m-n} is the $(m-n) \times 0$ matrix. If $m < n$, then let $M' = M \oplus Z'_{n-m}$ where Z'_{n-m} is the $0 \times (n-m)$ matrix. What this amounts to is either adding rows or columns of zeros as needed.

Now note that $\mathrm{sPf}([0]) = |0\rangle$. $\langle 0|$ is also a morphism in $\mathscr{P}$. Consider $\mathrm{sPf}^\vee(K) = \langle 0_A 0_B| + \langle 1_A 1_B|$ where

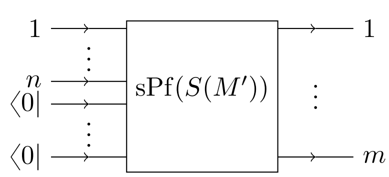
$$K = \begin{array}{c c} & \begin{array}{cc} A & B \end{array} \\ \begin{array}{c} A \\ B \end{array} & \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \end{array}$$

Contracting this with the morphism $|0_B\rangle$, we get $\langle 0_A|$.

Let M be an arbitrary $n \times m$ matrix. Then let us consider $S(M')$ where M' is defined as above. Suppose $n < m$. Then

$$\mathrm{sPf}(S(M')) = \sum_{I,J} \det(M_{I,J})|I 0_{n+1} \cdots 0_m\rangle\langle J|$$

Consider the following diagram in $\mathscr{P}$:



The morphism this represents will obviously come out to be $\mathrm{sDet}(M)$. If $n > m$, then copies of $|0\rangle$ are added to the extra output wires of $\mathrm{sPf}(S(M'))$. Thus we have finished the proof of theorem. Every morphism of $\mathcal{D}$ is in fact a morphism in $\mathscr{P}$. Furthermore, the reinterpretation of a determinantal circuit as a Pfaffian circuit can obviously be done in polynomial time. $\square$

Despite this fact, determinantal circuits still have some advantages. If a Pfaffian circuit can be represented as a determinantal circuit, its evaluation will be more efficient. Also, given the non-intuitive nature of the inclusion, viewing a circuit as determinantal may prove more useful than viewing it as a Pfaffian circuit. In the applications above, this is certainly true. Finally, the practical complexity of implementation is lessened by the determinantal approach since there is less need to track a complex ordering on the objects.

Acknowledgments

J.M. and J.T. were supported in part by the Defense Advanced Research Projects Agency under Award No. N66001-10-1-4040. Portions of J.T.'s work were sponsored by the Applied Research Laboratory's Exploratory and Foundational Research Program.

References

[1] V. Bergholm and J.D. Biamonte. Categorical quantum circuits. *Journal of Physics A: Math. Theor.*, 44:245304, 2011.

[2] W. Bruns, A. Conca, and M. Varbaro. Relations between the minors of a generic matrix. *Arxiv preprint arXiv:1111.7263*, 2011.

[3] A. Bulatov. The complexity of the counting constraint satisfaction problem. *Automata, Languages and Programming*, pages 646–661, 2010.

[4] J.Y. Cai and X. Chen. Complexity of counting csp with complex weights. In *Proceedings of the 44th symposium on Theory of Computing*, pages 909–920. ACM, 2012.

[5] F. R. K. Chung and Robert P. Langlands. A combinatorial Laplacian with vertex weights. *J. Combin. Theory Ser. A*, 75(2):316–327, 1996.

[6] C. Damm, M. Holzer, and P. McKenzie. The complexity of tensor calculus. *Computational Complexity*, 11(1):54–89, 2003.

[7] R.A. Horn and C.R. Johnson. *Matrix analysis*. Cambridge University Press, 1990.

[8] A. Joyal and R. Street. The geometry of tensor calculus. I. *Advances in Mathematics*, 88(1):55–112, 1991.

[9] A. Joyal, R. Street, and D. Verity. Traced monoidal categories. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 119, pages 447–468. Cambridge University Press, 1996.

[10] C. Kassel. *Quantum groups*, volume 155. Springer, 1995.

[11] Y. Lafont. Towards an algebraic theory of Boolean circuits. *Journal of Pure and Applied Algebra*, 184(2):257–310, 2003.

[12] JM Landsberg, J. Morton, and S. Norine. Holographic algorithms without matchgates. *Linear Algebra and its Applications*, 438(15), 2013.

[13] S. Lloyd, L. Maccone, R. Garcia-Patron, V. Giovannetti, Y. Shikano, S. Pirandola, L.A. Rozema, A. Darabi, Y. Soudagar, L.K. Shalm, and A. Steinberg. Closed timelike curves via postselection: theory and experimental test of consistency. *Physical Review Letters*, 106(4):40403, 2011.

[14] S. Mac Lane. *Categories for the working mathematician*. Springer verlag, 1998.

[15] J. Morton. Pfaffian circuits. *Arxiv preprint arXiv:1101.0129*, 2010.

[16] J. Morton and J. Biamonte. Undecidability in tensor network states. *Physical Review A*, 86(3):030301, 2012.

[17] K.D. Mulmuley and M. Sohoni. Geometric complexity theory i: An approach to the P vs. NP and related problems. *SIAM Journal on Computing*, 31(2):496–526, 2001.

[18] P. Selinger. A survey of graphical languages for monoidal categories. *New Structures for Physics*, pages 275–337, 2009.

[19] L. Valiant. Quantum circuits that can be simulated classically in polynomial time. *SIAM J. Comput.*, 31(4):1229–1254, 2002.